

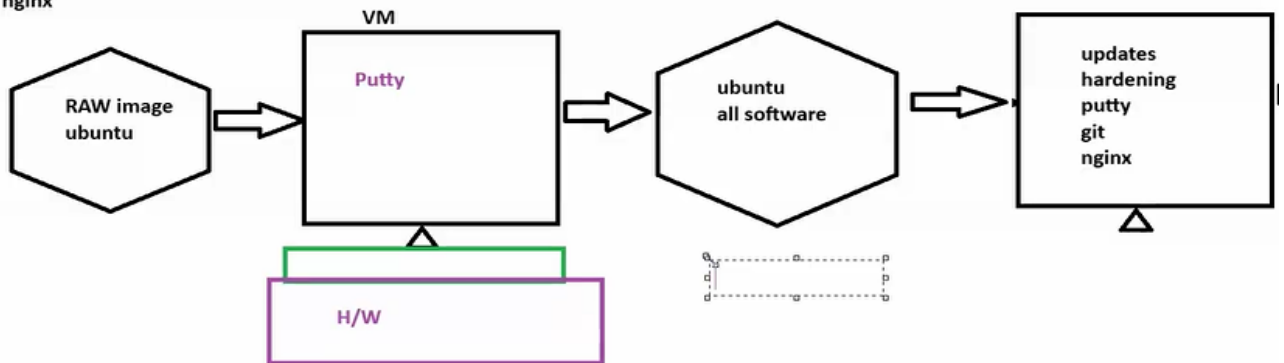
Image Creation

Project: 100 vm's

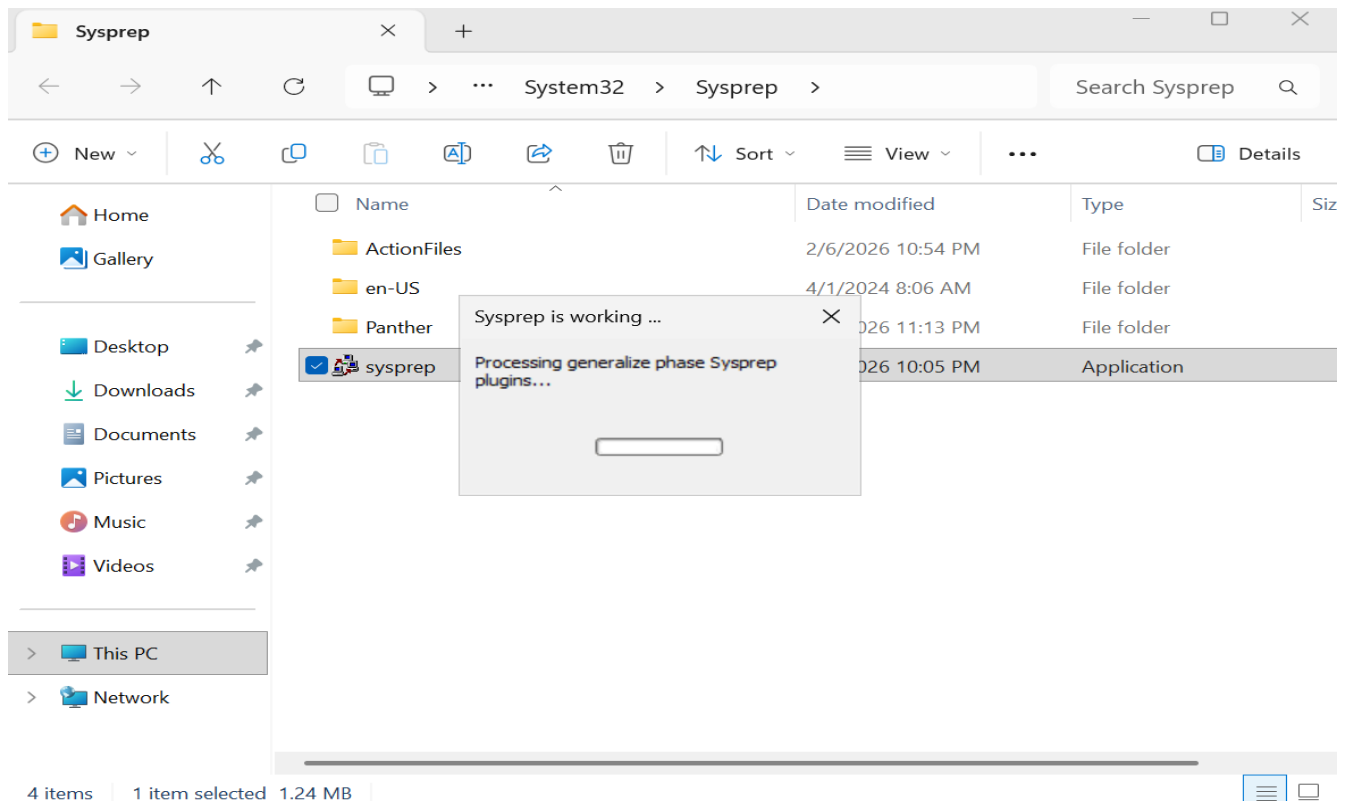
manually : Windows

updates
hardening
putty
git
nginx

automation : Packer tool : Linux



- ✓ **Create Image Manually using windows**
- ✓ **Next login it and download any software (Putty) and install it.**
- ✓ **Next search for Run to shut down hardware components for custom image.**
- ✓ **Run → sysprep → select sysprep (system preparation tool) → shutdown**



- ✓ **Next go to Vm on portal we can see Vm from Running to stopped.**
- ✓ **Next click on Capture on same Vm top middle → select Create image**

Create an image ...

Subscription ▼ Azure subscription 1

Resource group * ▼ Urgo1

Instance details

Region ▼ (US) Central US


Share image to Azure compute gallery ⓘ ☒ Yes, share it to a gallery as a VM image version.
☐ No, capture only a managed image.

Automatically delete this virtual machine after creating the image ⓘ ☒

Gallery details

Target Azure compute gallery * ⓘ ▼ (new) Imagegallery
[Create new](#)

Operating system state ⓘ ☒ Generalized: VMs created from this image require hostname, admin user, and other VM related setup to be completed on first boot
☐ Specialized: VMs created from this image are completely configured and do not require parameters such as hostname and admin user/password

 Capturing a virtual machine image will make the virtual machine unusable. This action cannot be undone.

Target VM image definition * ⓘ ▼ (new) Vmdef
[Create new](#)

Version details

Version number * ⓘ ▼ 01.22.33

[Review + create](#) [< Previous](#) [Next : Tags >](#)

✓ **Custom image** is created **Manually** by using Windows below.

Home > Microsoft.Compute-CaptureVM-20260211161502 | Overview

 **01.22.33 (Imagegallery/Vmdef/01.22.33)**
VM image version



How do I troubleshoot issues with this VM image version?






What are the metrics for this VM image version?

+1



  [+ Create VM](#) [+ Create VMSS](#) [Delete](#) [Refresh](#) [Give feedback](#)

Overview

-  Activity log
-  Access control (IAM)
-  Tags
-  Diagnose and solve problems
-  Resource visualizer
- > Settings
- > Automation
- > Help

Essentials

Resource group [\(move\)](#) : [Urgo1](#)

Status : Succeeded

Location : Central US

Subscription [\(move\)](#) : [Azure subscription 1](#)

Subscription ID : 5d75b66d-66bf-44e0-8d7e-7e61e4b043d7

Azure compute gallery : [Imagegallery](#)

VM image definition : [Vmdef](#)

Replication status : Completed

Replication mode : Full

Confidential OS disk encr... : -

Encryption type : Platform-managed key

End of life date : -

Exclude from latest : No

Lock deleting Replicated ... : Yes

Storage account type : Standard HDD LRS

[JSON View](#)

Tags [\(edit\)](#) : [Add tags](#)

- Let's **automate** same process by using **Packer tool** and create Custom image of **Linux machine**

```
Windows PowerShell

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\packer> code .
PS C:\packer> .\packer.exe --version
Packer v1.15.0
PS C:\packer> .\packer.exe -- version
Usage: packer [--version] [--help] <command> [<args>]

Available commands are:
  build      build image(s) from template
  console    creates a console for testing variable interpolation
  fix        fixes templates from old versions of packer
  fmt        Rewrites HCL2 config files to canonical format
  hcl2_upgrade transform a JSON template into an HCL2 configuration
  init       Install missing plugins or upgrade plugins
  inspect    see components of a template
  plugins    Interact with Packer plugins and catalog
  validate   check that a template is valid
  version    Prints the Packer version
```

✓ **Code .** it will navigate to VS Code

```
File Edit Selection View ... packer

EXPLORER
  PACKER
    packer.exe
    packer.json

packer.json
1
2
3 "builders": [{
4   "type": "azure-arm",
5
6   "client_id": "00001111-aaaa-2222-bbbb-3333cccc4444",
7   "client_secret": "0e760437-bf34-4aad-9f8d-870be799c55d",
8   "tenant_id": "aaaabbbb-0000-cccc-1111-dddd2222eeee",
9   "subscription_id": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
10
11   "managed_image_resource_group_name": "myResourceGroup",
12   "managed_image_name": "myPackerImage",
13
14   "os_type": "linux",
15   "image_publisher": "canonical",
16   "image_offer": "0001-com-ubuntu-server-jammy",
17   "image_sku": "22_04-lts",
18
19   "azure_tags": {
20     "dept": "Engineering",
21     "task": "Image deployment"
22   },
23
24   "location": "East US",
25   "vm_size": "Standard_DS2_v2"
26 },]
27
28 "provisioners": [{
29   "execute_command": "chmod +x {{ .Path }}; {{ .Vars }} sudo -E sh '{{ .Path }}'",
30   "inline": [
31     "apt-get update",
32     "apt-get upgrade -y",
33     "apt-get -y install nginx",
34
35     "/usr/sbin/waagent -force -deprovision+user && export HISTSIZE=0 && sync"
36   ],
37   "inline_shebang": "/bin/sh -x",
38   "type": "shell"
39 },]
```

- ✓ In Vs code there is **folder packer.exe**
- ✓ Click on it and **choose new file** and give name **packer.json**
- ✓ File should be in Json format only.
- ✓ Next for image creation code go to below and choose Linux format



Microsoft Learn

<https://learn.microsoft.com> › azure › linux › imaging

Overview of creating Linux images for Azure

Create an image that can be configured for use by multiple VMs. You can set the hostname, add an admin user, and perform other tasks during first boot. [Read more](#)

- ✓ **Copy Linux code and paste on VS code**

```
packer => image creation

to create service account

navigate to Azure cloud => search for Microsoft entra id => manage => app registration => New registration
```

- ✓ Next Create **service account** for that follow steps above
- ✓ After creating account, we will get below details
- ✓ Choose **client_id**, **client_secret**, **tenant_id**, **subscription_id** copy these from below portal and paste in VS code.

The screenshot shows the Microsoft Entra ID portal for an application named 'udayb30'. The left sidebar contains navigation links: Overview, Quickstart, Integration assistant, Diagnose and solve problems, Manage, and Support + Troubleshooting. The main content area displays the 'Essentials' section with the following details:

Display name	: udayb30	Client credentials	: Add a certificate or secret
Application (client) ID	: 0ff1669f-dc34-47d9-a293-3ea4964e9015	Redirect URIs	: Add a Redirect URI
Object ID	: 709d559d-86f8-4c0f-b1d3-d115ea04ad00	Application ID URI	: Add an Application ID URI
Directory (tenant) ID	: c222eaff-4ca0-4dbc-912a-df0739bcb6f5	Managed application in I...	: udayb30
Supported account types	: My organization only		

- ✓ **Shown below**

```
{
  "client_id": "0ff1669f-dc34-47d9-a293-3ea4964e9015",
  "client_secret": "YCV8Q~aoJXXDq1LipYcLEPYusC3UgNZM-PkMjbFa",
  "tenant_id": "c222eaff-4ca0-4dbc-912a-df0739bcb6f5",
  "subscription_id": "5d75b66d-66bf-44e0-8d7e-7e61e4b043d7",
}
```

- ✓ For **client_secret** go to **manage** → **certificates & secrets** follow below.

✓ Client secret id below (Copy Value)

🔑 udayb30 | Certificates & secrets ✨ ...

Search

Got feedback?

Got a second to give us some feedback? →

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) **Client secrets (1)** Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value
secret	10/08/2026	YCV8Q~aoJXxDq1LipYcLEPYusC3UgNZ... 5dfe5b10-3054-4239-b6f6-c3021d73c06a

Copied secret ID

✓ Next our free subscription ID copy and paste.

➤ Right now, **Authentication** is completed

- But **Udayb30** user need **Authorisation** on **subscription**, for that we should give some role like Owner or etc...

✓ Go to IAM (Identity Access Management) below

Home > Subscriptions > Azure subscription 1

Subscriptions << >>

Default Directory (reddylkp2004@gmail.onmicrosoft...)

+ Add Manage Policies ...

Global administrators can manage all subscriptions in this list by updating their policy setting [here](#).

View list of subscriptions for which you have role-based access control (RBAC) permissions to manage Azure resources. To view subscriptions for which you have billing access, [click here](#).

Showing subscriptions in Default Directory directory. Don't see a subscription? [Switch directories](#)

Subscriptions: Filtered (1 of 1)

My role == all

Status == all

+ Add filter

Subscription name ↑

Azure subscription 1 | Access control (IAM) ☆ ...

Subscription

Search

+ Add ↓ Download role assignments Edit columns Refresh Delete Feedback

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Security

Resource visualizer

Events

Resource groups

Resources

Cost Management

Cost analysis

Action required: As of August 31, 2024, [Azure classic administrator roles](#) (along with Azure classic resources, Azure Service Manager) are retired and are no longer supported. If you still have active Co-Administrator or Service Administrator role assignments, convert these role to Azure RBAC immediately. [Learn more](#)

Check access Role assignments Roles Deny assignments Classic administrators

My access

View my level of access to this resource.

View my access

Check access

Review the level of access a user, group, service principal, or managed identity has to this resource. [Learn more](#)

Check access



Add



Download role assignments



Overview



Activity log



Access control (IAM)



Tags

Add role assignment

Add co-administrator

Add custom role

of August 31, 2024, Azure c
onger supported. If you still
mediately. [Learn more](#)

Add role assignment

Role

Members

Conditions

Review + assign

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. [Learn more](#)



Copilot can help pick a role

Job function roles

Privileged administrator roles

Grant privileged administrator access, such as the ability to assign roles to other users.



Can a job function role with less access be used instead?



Search by role name, description, permission, or ID

Type : All

Category : All

Name



Description



Owner

Grants full access to manage all resources, including the ability to assign r

Contributor

Grants full access to manage all resources, but does not allow you to assign

Access Review Operator Service Role

Lets you grant Access Review System app permissions to discover and rev

Azure IoT Operations Onboarding

User can Azure arc connect and deploy Azure IoT Operations securely.

Azure Resilience Management Drills Admini...

Administrator Role of Azure Resilience Management Drills Service

Review + assign

Previous

Next

✓ Below added user **Udayb30** and I want to give **owner role**

Role

Members

Conditions

Review + assign

Selected role

Owner

Assign access to



User, group, or service principal



Managed identity

Members

+ Select members

Name

Object ID

Type

udayb30

f28ed8b3-ef51-414b-bb06-69eff71a4b96 App

Description

Optional

Role Members Conditions Review + assign

Selected role

Owner

What user can do

- ☐ Allow user to only assign selected roles to selected principals (fewer privileges) ⓘ
- ☐ Allow user to assign all roles except privileged administrator roles Owner, UAA, RBAC (Recommended) ⓘ
- ☒ Allow user to assign all roles (highly privileged) ⓘ

⚠ Owner is a privileged admin role that grants privileged administrator access, such as the ability to assign roles to ot narrow the permissions of this role to least privilege.

✓ **To run packer, we need Authentication with authorisation then only we can execute, we have done all below.**

```
· "client_id": "0ff1669f-dc34-47d9-a293-3ea4964e9015",  
· "client_secret": "YCV8Q~aoJXxDq1LipYclEPYusC3UgNZM-PkMjbFa",  
· "tenant_id": "c222eaff-4ca0-4dbc-912a-df0739bcb6f5",  
· "subscription_id": "5d75b66d-66bf-44e0-8d7e-7e61e4b043d7",
```

✓ **Below give**

- **Resource group name**
- **Image name**
- **OS type**
- **Location**
- **VM size**

```
Welcome {} packer.json
{} packer.json > [] builders > {} 0 > abc vm_size
2   "builders": [{
3     "type": "azure-arm",
4
5     "client_id": "0ff1669f-dc34-47d9-a293-3ea4964e9015",
6     "client_secret": "YCV8Q~aoJXxDq1LipYclEPYusC3UgNZM-PkMjbFa",
7     "tenant_id": "c222eaff-4ca0-4dbc-912a-df0739bcb6f5",
8     "subscription_id": "5d75b66d-66bf-44e0-8d7e-7e61e4b043d7",
9
10    "managed_image_resource_group_name": "Uday-Rg",
11    "managed_image_name": "myudaylinux-Image",
12
13    "os_type": "Linux",
14    "image_publisher": "canonical",
15    "image_offer": "0001-com-ubuntu-server-jammy",
16    "image_sku": "22_04-lts",
17
18    "azure_tags": {
19      "dept": "Engineering",
20      "task": "Image deployment"
21    },
22
23    "location": "East US",
24    "vm_size": "Standard_B1s"
25  }],
26  "provisioners": [{
27    "execute_command": "chmod +x {{ .Path }}; {{ .Vars }} sudo -E sh '{{ .Path }}'",
28    "inline": [
29      "apt-get update",
30      "apt-get upgrade -y",
31      "apt-get -y install nginx",
32
33      "/usr/sbin/waagent -force -deprovision+user && export HISTSIZE=0 && sync"
34    ],
35    "inline_shebang": "/bin/sh -x",
36    "type": "shell"
37  ]
}
```

- ✓ Make sure everything is done save first **Ctrl+s**
- ✓ Then open command prompt and execute below

```
PS C:\packer> .\packer.exe inspect .\packer.json
Packer Inspect: JSON mode
Variables:

<No variables>

Builders:

  azure-arm

Provisioners:

  shell

Note: If your build names contain user variables or template
functions such as 'timestamp', these are processed at build time,
and therefore only show in their raw form here.
PS C:\packer>
```


- ✓ Below is complete code from **start to build**

```
Windows PowerShell

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\packer> code .
PS C:\packer> .\packer.exe --version
Packer v1.15.0
PS C:\packer> .\packer.exe -- version
Usage: packer [--version] [--help] <command> [<args>]

Available commands are:
  build          build image(s) from template
  console        creates a console for testing variable interpolation
  fix            fixes templates from old versions of packer
  fmt           Rewrites HCL2 config files to canonical format
  hcl2_upgrade   transform a JSON template into an HCL2 configuration
  init          Install missing plugins or upgrade plugins
  inspect       see components of a template
  plugins       Interact with Packer plugins and catalog
  validate      check that a template is valid
  version       Prints the Packer version

PS C:\packer> .\packer.exe inspect .\packer.json
Packer Inspect: JSON mode
Variables:

<No variables>

Builders:

  azure-arm

Provisioners:

  powershell

Note: If your build names contain user variables or template
functions such as 'timestamp', these are processed at build time,
and therefore only show in their raw form here.
PS C:\packer> .\packer.exe build .\packer.json
azure-arm: output will be in this color.

==> azure-arm: Running builder ...
==> azure-arm: Creating Azure Resource Manager (ARM) client ...
==> azure-arm: ARM Client successfully created
```

- ✓ For the first time before **packer build**, we should install Plugin.
- ✓ If we don't it asks to install plugin, then enter below
- ✓ **./packer plugins install github.com/hashicorp/azure**

```
$ packer plugins install github.com/hashicorp/azure
```

- ✓ Next run build command, packages will build automatically and custom image we see in resource group (**Uday-Rg**).

✓ Image is Created.

```
Windows PowerShell
==> azure-arm: Deleting Virtual Machine deployment and its attached resources...
==> azure-arm: Deleted -> Microsoft.Compute/virtualMachines : 'pkrvmqh3j2lxjn7'
==> azure-arm: Deleted -> Microsoft.Network/networkInterfaces : 'pkrnqiqh3j2lxjn7'
==> azure-arm: Deleted -> Microsoft.Network/virtualNetworks : 'pkrvnqh3j2lxjn7'
==> azure-arm: Deleted -> Microsoft.Network/publicIPAddresses : 'pkripqh3j2lxjn7'
==> azure-arm: Deleted -> Microsoft.Network/networkSecurityGroups : 'pkrsqqh3j2lxjn7'
==> azure-arm: Deleted -> Microsoft.Compute/disks : '/subscriptions/5d75b66d-66bf-44e0-8d7e-7e61e4b043d7/resourceGroups/UcentralUSRg/providers/Microsoft.Compute/disks/pkrosqh3j2lxjn7'
==> azure-arm: Removing the created Deployment object: 'pkrdpqh3j2lxjn7'
==> azure-arm:
==> azure-arm: Deleting KeyVault created during build
==> azure-arm: Deleted -> Microsoft.KeyVault/vaults : 'pkrvqh3j2lxjn7'
==> azure-arm: Removing the created Deployment object: 'kvpkrdpqh3j2lxjn7'
==> azure-arm:
==> azure-arm: The resource group was not created by Packer, not deleting ...
Build 'azure-arm' finished after 10 minutes 37 seconds.

==> Wait completed after 10 minutes 38 seconds

==> Builds finished. The artifacts of successful builds are:
--> azure-arm: Azure.ResourceManagement.VMImage:

OSType: Windows
ManagedImageResourceGroupName: UcentralUSRg
ManagedImageName: UdayPackerImage
ManagedImageId: /subscriptions/5d75b66d-66bf-44e0-8d7e-7e61e4b043d7/resourceGroups/UcentralUSRg/providers/Microsoft.Compute/images/UdayPackerImage
ManagedImageLocation: centralus

PS C:\packer>
```



➤ Automate Windows (Image Creation)

```
packer.exe  packer.json X
packer.json > [] builders > {} 0
1  {
2    "builders": [{
3      "type": "azure-arm",
4
5      "client_id": "bbd8d2fa-5fc7-40f5-97a4-ea93ed316618",
6      "client_secret": "IxS8Q~hzGAvegg_Bkx1U02nCXAg09MJge4mapbZP",
7      "tenant_id": "c222eaff-4ca0-4dbc-912a-df0739bcb6f5",
8      "subscription_id": "5d75b66d-66bf-44e0-8d7e-7e61e4b043d7",
9
10     "managed_image_resource_group_name": "UcentralUSRg",
11     "managed_image_name": "UdayPackerImage",
12
13     "os_type": "Windows",
14     "image_publisher": "MicrosoftWindowsServer",
15     "image_offer": "WindowsServer",
16     "image_sku": "2016-Datacenter",
17
18     "communicator": "winrm",
19     "winrm_use_ssl": true,
20     "winrm_insecure": true,
21     "winrm_timeout": "5m",
22     "winrm_username": "packer",
23
24     "azure_tags": {
25       "dept": "Engineering",
26       "task": "Image deployment"
27     },
28
29     "build_resource_group_name": "UcentralUSRg",
30     "vm_size": "Standard_D2s_v3"
31   }],
32   "provisioners": [{
33     "type": "powershell",
34     "inline": [
35       "Add-WindowsFeature Web-Server",
36       "while ((Get-Service RdAgent).Status -ne 'Running') { Start-Sleep -s 5 }",
37       "while ((Get-Service WindowsAzureGuestAgent).Status -ne 'Running') { Start-Sleep -s 5 }",
38       "& $env:SystemRoot\\System32\\Sysprep\\Sysprep.exe /oobe /generalize /quiet /quit",
39       "while($true) { $imageState = Get-ItemProperty HKLM:\\SOFTWARE\\Microsoft\\Windows\\Current
40     ]
41   }]
42 }
```

```
Windows PowerShell

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\packer> code .
PS C:\packer> .\packer.exe --version
Packer v1.15.0
PS C:\packer> .\packer.exe -- version
Usage: packer [--version] [--help] <command> [<args>]

Available commands are:
  build      build image(s) from template
  console    creates a console for testing variable interpolation
  fix        fixes templates from old versions of packer
  fmt        Rewrites HCL2 config files to canonical format
  hcl2_upgrade transform a JSON template into an HCL2 configuration
  init       Install missing plugins or upgrade plugins
  inspect    see components of a template
  plugins    Interact with Packer plugins and catalog
  validate   check that a template is valid
  version    Prints the Packer version

PS C:\packer> .\packer.exe inspect .\packer.json
Packer Inspect: JSON mode
Variables:

  <No variables>

Builders:

  azure-arm

Provisioners:

  powershell

Note: If your build names contain user variables or template
functions such as 'timestamp', these are processed at build time,
and therefore only show in their raw form here.
PS C:\packer> .\packer.exe build .\packer.json
azure-arm: output will be in this color.

==> azure-arm: Running builder ...
==> azure-arm: Creating Azure Resource Manager (ARM) client ...
==> azure-arm: ARM Client successfully created
```

- ✓ Now we no need to install plugin just run build package will be created.

```
Windows PowerShell
==> azure-arm: Deleting Virtual Machine deployment and its attached resources...
==> azure-arm: Deleted -> Microsoft.Compute/virtualMachines : 'pkrvmqh3j2lxjn7'
==> azure-arm: Deleted -> Microsoft.Network/networkInterfaces : 'pkrnigh3j2lxjn7'
==> azure-arm: Deleted -> Microsoft.Network/virtualNetworks : 'pkrvnqh3j2lxjn7'
==> azure-arm: Deleted -> Microsoft.Network/publicIPAddresses : 'pkripqh3j2lxjn7'
==> azure-arm: Deleted -> Microsoft.Network/networkSecurityGroups : 'pkrsqgh3j2lxjn7'
==> azure-arm: Deleted -> Microsoft.Compute/disks : '/subscriptions/5d75b66d-66bf-44e0-8d7e-7e61e4b043d7/resourceGroups/UcentralUSRg/providers/Microsoft.Compute/disks/pkrosqh3j2lxjn7'
==> azure-arm: Removing the created Deployment object: 'pkrdpqh3j2lxjn7'
==> azure-arm:
==> azure-arm: Deleting KeyVault created during build
==> azure-arm: Deleted -> Microsoft.KeyVault/vaults : 'pkrkvqh3j2lxjn7'
==> azure-arm: Removing the created Deployment object: 'kvpkrdpqh3j2lxjn7'
==> azure-arm:
==> azure-arm: The resource group was not created by Packer, not deleting ...
Build 'azure-arm' finished after 10 minutes 37 seconds.

==> Wait completed after 10 minutes 38 seconds

==> Builds finished. The artifacts of successful builds are:
--> azure-arm: Azure.ResourceManagement.VMImage:

OSType: Windows
ManagedImageResourceGroupName: UcentralUSRg
ManagedImageName: UdayPackerImage
ManagedImageId: /subscriptions/5d75b66d-66bf-44e0-8d7e-7e61e4b043d7/resourceGroups/UcentralUSRg/providers/Microsoft.Compute/images/UdayPackerImage
ManagedImageLocation: centralus

PS C:\packer>
```